Pattern Recognition

Assignment -1

Report

Name - Urvashi Goswami
Roll No – S22024
Course – M. Tech (R)

Index

Colab Links

Question 1. Univariate Speech Activity Detection

https://colab.research.google.com/drive/1UvBuMo7gKq8i-Uew2pYbEcgPW4uuhLhq?usp=sharing

Question 2 Linearly Separable Case

https://colab.research.google.com/drive/1zUesv36WGu7X4esSUXRJbO9a7Fj9ikbj?usp=sharing

Question 2 Non-Linearly Separable Case

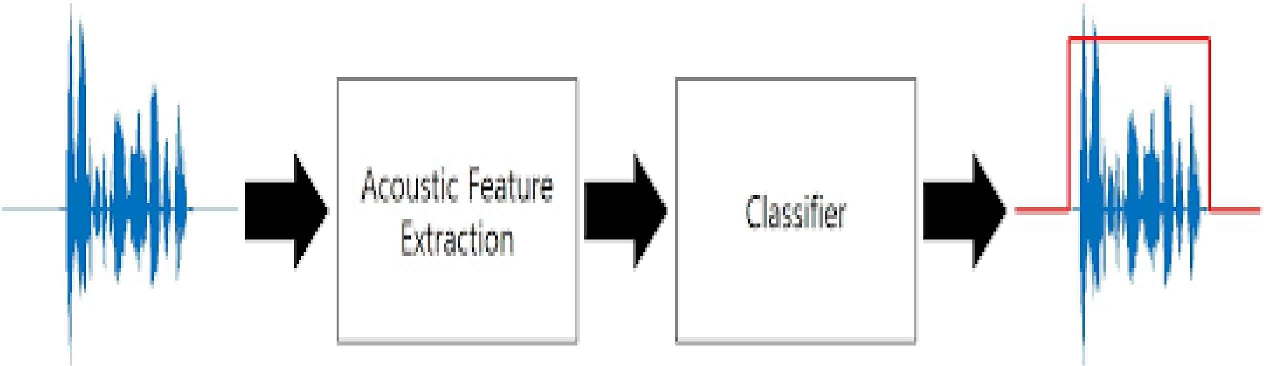
https://colab.research.google.com/drive/19bihvFhGwiAUkFFQtZvaBgQopUW5G3R5?usp=sharing

Question 1. You are going to perform speech activity detection (SAD.) Given a sequence of signal frames, classify each frame as speech or non-speech. Two types of 1-D features are provided: short-time energy, and Mel-filterbank energy. Which of these features are better at correctly detecting speech? Plot ROC curves to justify your choice.

- **You can use a simple unimodal Gaussian to estimate the distribution of the features. Use sample mean and sample variance as parameters of the Gaussian.**
- **The ground truth _les are provided with 1 meaning speech and 0 meaning non-speech.**
- **Use Segment 2 for estimating the model and segment 3 for testing (i.e., ROC curves will be computed on Segment 3.)**

Expected outputs: **Plot of ROC curves for each feature used.**

Speech activity detection (**SAD**), also known as **voice activity detection** or **speech detection**, is the detection of the presence or absence of human speech, used in speech processing. The main uses of SAD are in speech coding and speech recognition. It can facilitate speech processing and can also be used to deactivate some processes during non-speech section of an audio session.



Steps in Speech Detection:

Step1: Speech must be converted from a physical sound to an electrical signal with a microphone, and then to digital data with an analog-to-digital converter. i.e., Data Collection

Step2: Data Pre-processing

Step3: Feature extraction

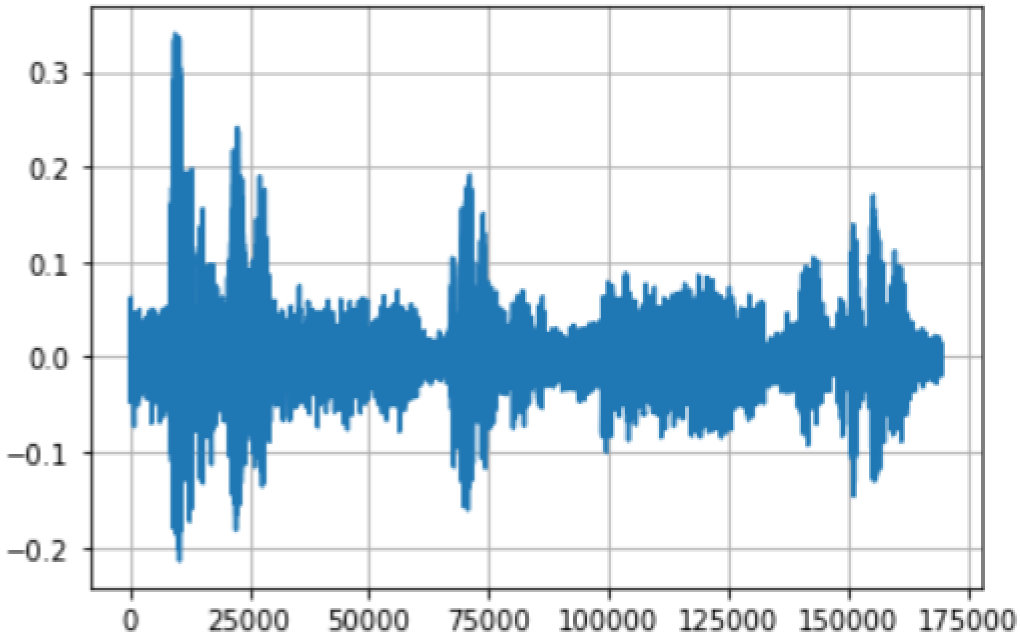
Step4: Training classifier

Step5: Testing classifier

Step6: Performance Check using Receiver Operating Characteristic Curve (ROC)

Since Step1 and Step2 are already done in our case. We need to directly train our classifier. We will use simple **unimodal Gaussian distribution** to estimate the distribution of the features.

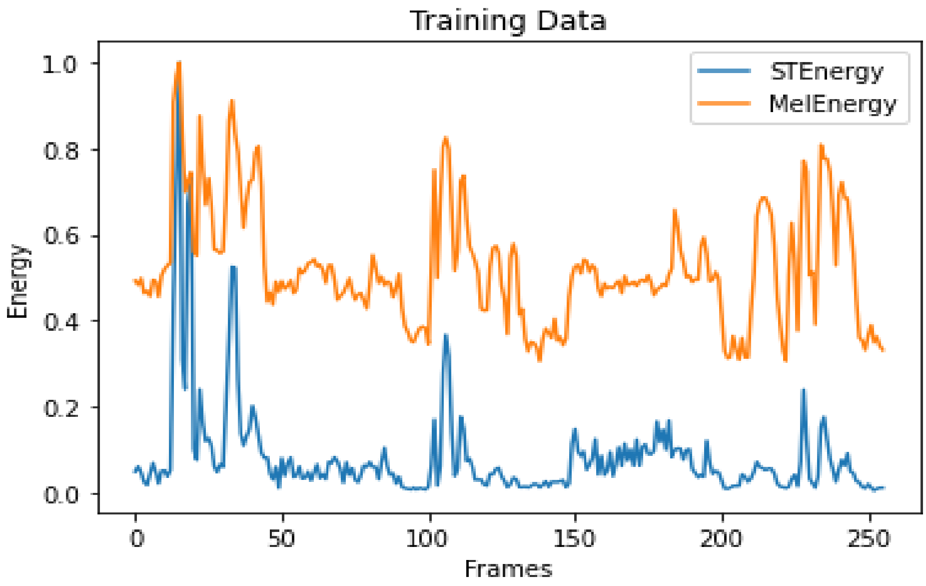
Plotting sample audio using **librosa ML library** to get a visualization of sample audio signal.

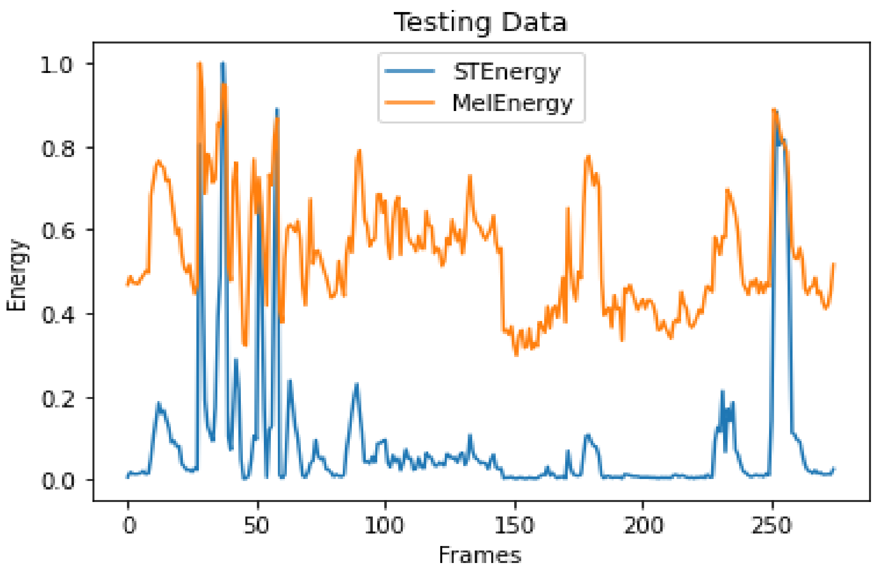


In our case, two pre-processed 1D features are already given to us.

- **Short Time Energy**
- **Mel-Filterbank Energy**

Plotting our given feature data of 255 sample frames, we get the following plots:

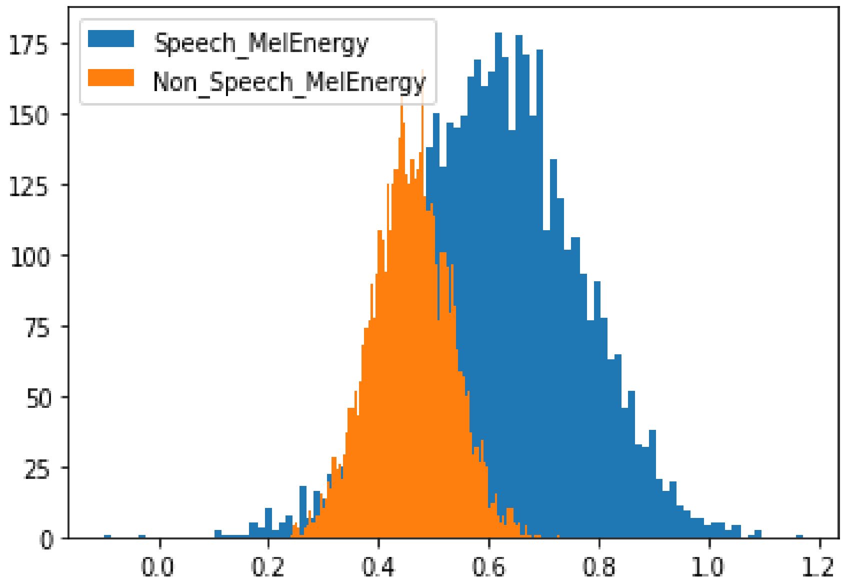




We split our data into speech and non-speech for both MelEnergy and STEnergy using Ground Truth values and plotting normal distribution histogram we get the following distribution graph:

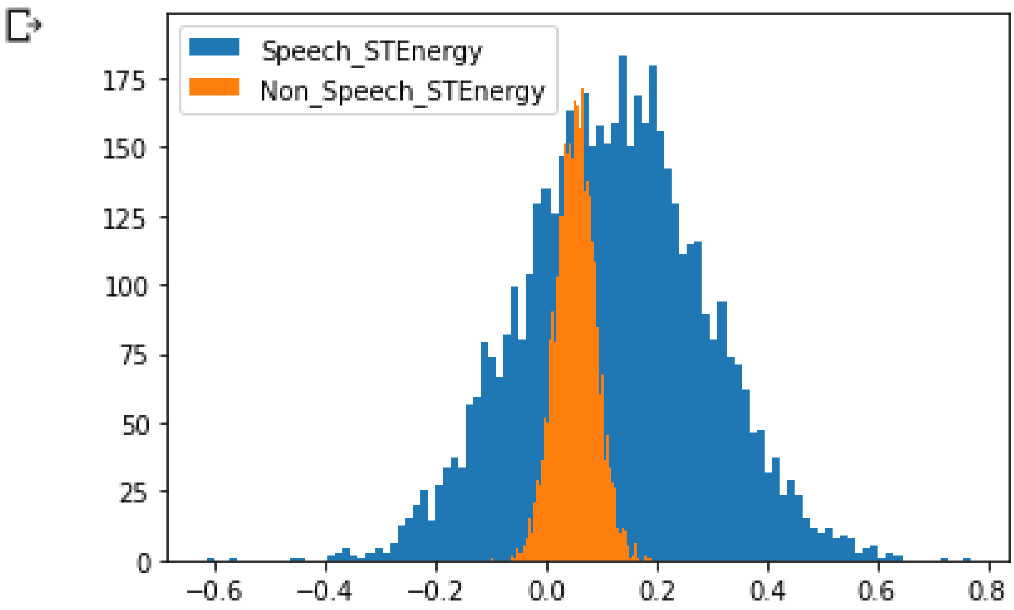
MelEnergy

- **Speech Frames:** 111
- **Non-Speech Frames:** 145
- **Mean_speech_seg2_MELEnergy:** 0.6100466456738739
- **Mean_non_speech_seg2_MELEnergy:** 0.45865463603448275
- **std_speech_seg2_MELEnergy:** 0.15122270736844518
- **std_non_speech_seg2_MELEnergy:** 0.07200991698472241



STEnergy

- **Speech Frames:** 111
- **Non-Speech Frames:** 145
- **Mean_speech_seg2_STEnergy:** 0.12043432332432433
- **Mean_non_speech_seg2_STEnergy:** 0.053090674108275865
- **std_speech_seg2_STEnergy:** 0.16637685909788924
- **std_non_speech_seg2_STEnergy:** 0.03597781981881293



Normal Gaussian Distribution:

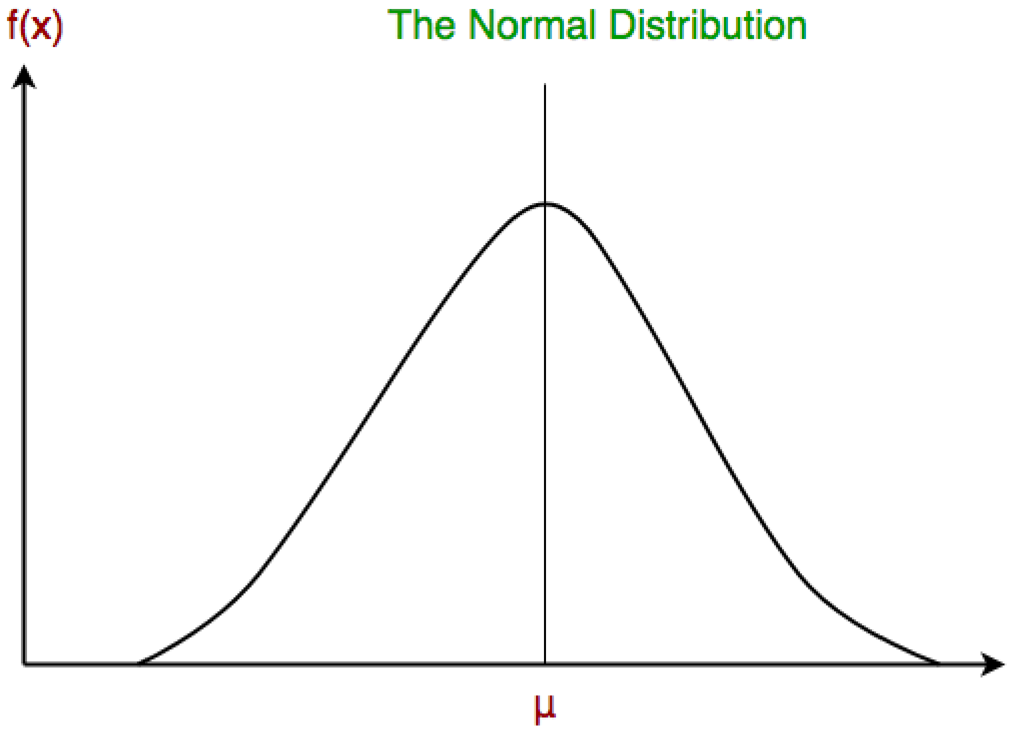
Normal distribution is the default probability for many real-world scenarios. It represents a symmetric distribution where most of the observations cluster around the central peak called as mean of the distribution. A normal distribution can be thought of as a bell curve or Gaussian Distribution which typically has two parameters: mean and standard deviation (SD). The parameter used to measure the variability of observations around the mean is called standard deviation. The probabilities for values occurring near the mean are higher than the values far away from the mean. The parameters of the normal distribution plot defining the shape and the probabilities are mean and standard deviation. The area of the plot between two different points in the normal distribution plot represents the probability of the value occurring between those two points.

Here is the **probability density function** for normal distribution:

$$F(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2 / 2\sigma^2}$$

Where, x is the variable, mu is the mean, and sigma standard deviation

When plotted, it gives a bell-shaped curve which is symmetric about the mean of the feature values as shown below:



Bayes' Theorem

Bayes' Theorem finds the probability of an event occurring given the probability of another event that has already occurred. Bayes' theorem is stated mathematically as the following equation:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

where A and B are events and $P(B) \neq 0$.

- Basically, we are trying to find probability of event A, given the event B is true. Event B is also termed as **evidence**.
- P(A) is the **priori** of A (the prior probability, i.e., Probability of event before evidence is seen). The evidence is an attribute value of an unknown instance.

- P(A|B) is a **posteriori** probability of B, i.e., probability of event after evidence is seen.

Now, with regards to our dataset, we can apply Bayes' theorem in following way:

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)}$$

where, y is class variable and X is a dependent feature vector (of size *n*) where:

$$X = (x_1, x_2, x_3, ..., x_n)$$

P(Speech|xi) = P(xi|Speech) * P(speech) / P(xi)
P(xi) = P(xi|Speech) * P(Speech) + P(xi|Non-Speech) * P(Non-Speech)

P(Speech|xi): Posterior probability of speech samples given **xi** samples of test
P(xi|Speech): Likelihood
P(Speech): Priori of speech
P(xi): Evidence

Prior Probability:

Prior (Speech) = No. of Speech Frames / Total No. of Frames

Prior (Non-Speech) = No. of Non-Speech Frames / Total No. of Frames

- **prior_speech_seg2_MelE:** 0.43359375
- **prior_non_speech_seg2_MelE:** 0.56640625
- **prior_speech_seg2_STE:** 0.43359375
- **prior_non_speech_seg2_STE**: 0.56640625

ROC Curve

A Receiver Operator Characteristic (ROC) curve is a graphical plot used to show the diagnostic ability of binary classifiers. It was first used in signal detection theory but is now used in many other areas such as medicine, radiology, natural hazards and machine learning.

TP = True Positive Fraction (Sensitivity)= TP/ (TP+FN)

FN = False Negative Fraction (1-Sensitivity) = FN/ (TP+FN)

TN =True Negative Fraction (Specificity)= TN/ (TN+FP)

FP = False Positive Fraction (1-specificity) = FP/ (TN+FP)

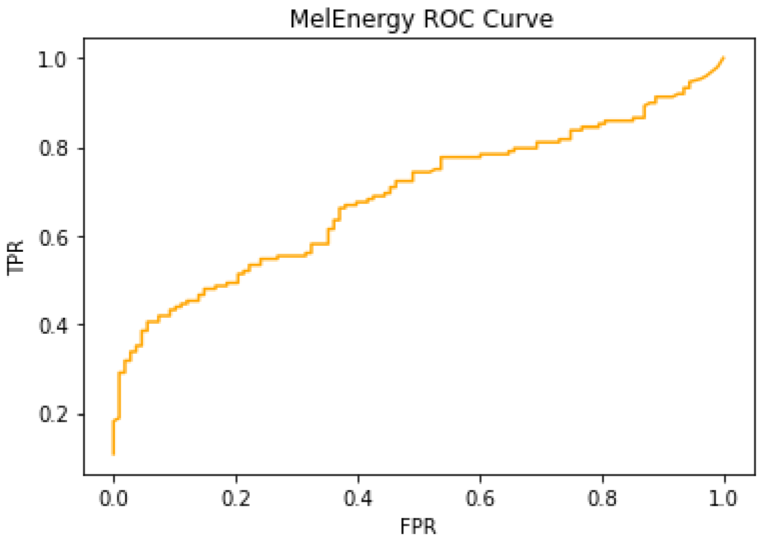
TPR = True Positive Rate = TP/(TP+FP)

FPR = False Negative Rate = TN/(TN+FN)

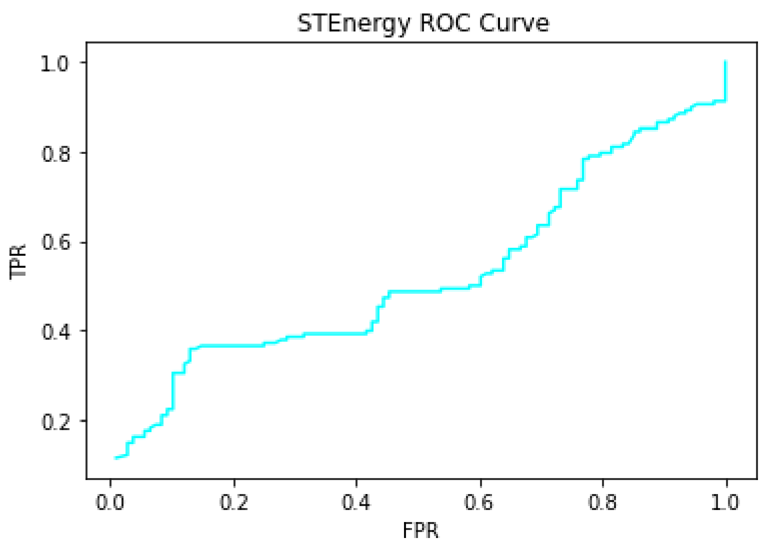
Creating a ROC curve

A ROC curve is constructed by plotting the true positive rate (TPR) against the false positive rate (FPR). The true positive rate is the proportion of observations that were correctly predicted to be positive out of all positive observations (TP/(TP + FN)). Similarly, the false positive rate is the proportion of observations that are incorrectly predicted to be positive out of all negative observations (FP / (TN + FP)).

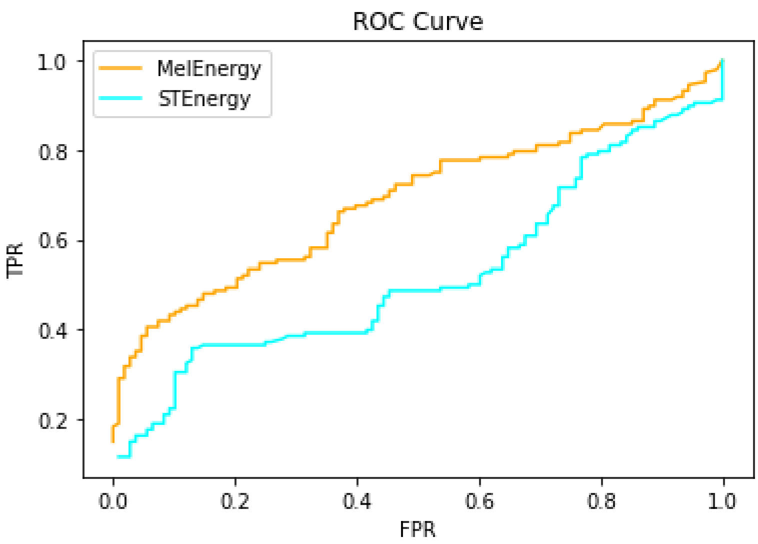
ROC Curve of MelEnergy



ROC Curve of STEnergy



Comparison of ROC Curves



Conclusion: MelEnergy Feature performs better job than STEnergy in classifying speech and non-speech signals.

Question 2. Develop a Bayes classifier with Gaussian class conditional densities to classify two datasets, each having 3 classes. The first dataset is linearly separable, and the second is not. Use random 50% of data for training and 50% for test. Build the following classifiers C1-C4:

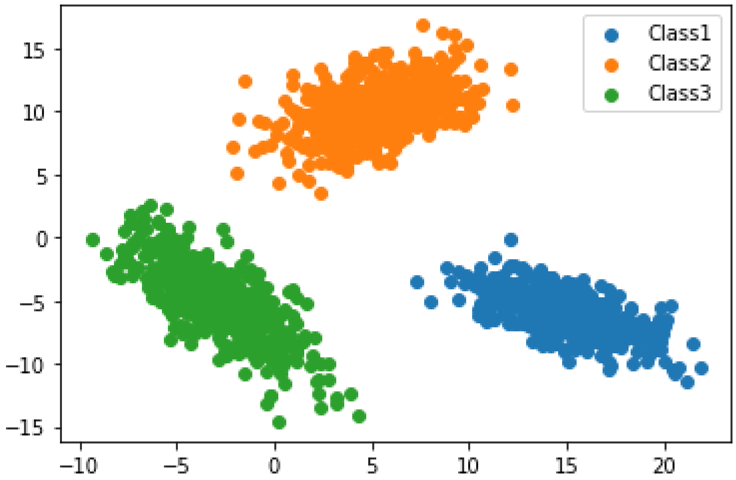
- **C1: Covariance for all classes is IE^2. Use the average of the sample variances for all dimensions, for all classes, from the training data as 2.**
- **C2: Full but equal covariance for all classes, _. Use the average of the sample covariance matrix from all classes in the train data.**
- **C3: Diagonal covariance matrix, distinct for each class. Use variances from the sample covariance matrix for each class.**
- **C4: Full covariance matrix, distinct for each class. Use the sample covariance matrix for each class.**

Expected Result:

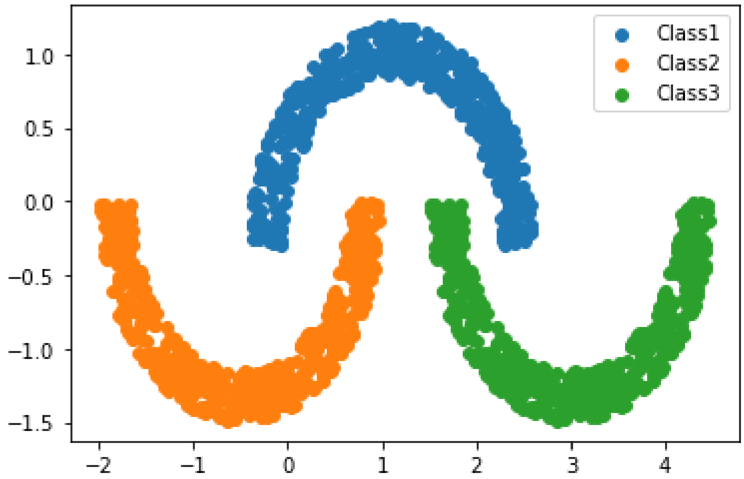
1. **Summarize the classifier performance as in Table 1. Use separate tables for linear and non-linear data.**
2. **For each classifier, and for each dataset, plot the decision regions with class data in different colours.**

According to our question, we will split our training and testing data as 50% - 50% i.e., 250-250 number of samples in each class.

Plotting Linearly Separable Data using Scatter Plot, we get the following:



Plotting Non-Linearly Separable Data using Scatter Plot, we get the following plot



Mean of Training Data:

Mean of Training Data Class 1	[14.9438104 -6.0137616]
Mean of Training Data Class 2	[5.07319964 9.9348812]
Mean of Training Data Class 3	[-3.07815664 -4.96572593]

Variance of Training Data:

Variance of Training Data Class 1	[6.02755383 2.77801584]
Variance of Training Data Class 2	[6.49502067 4.77123273]
Variance of Training Data Class 3	[4.78847509 9.7383084]

Covariance of Training Data:

Covariance of Training Data Class 1	[[6.05176087 -2.65004037] [-2.65004037 2.78917253]]
Covariance of Training Data Class 2	[[6.52110509 2.94634999] [2.94634999 4.79039431]]
Covariance of Training Data Class 3	[[4.80770592 -5.07829604] [-5.07829604 9.77741807]]

Prior of Training Data:

Prior of Training Data Class 1	0.3333333
Prior of Training Data Class 2	0.3333333
Prior of Training Data Class 3	0.3333333

Multivariate Distribution

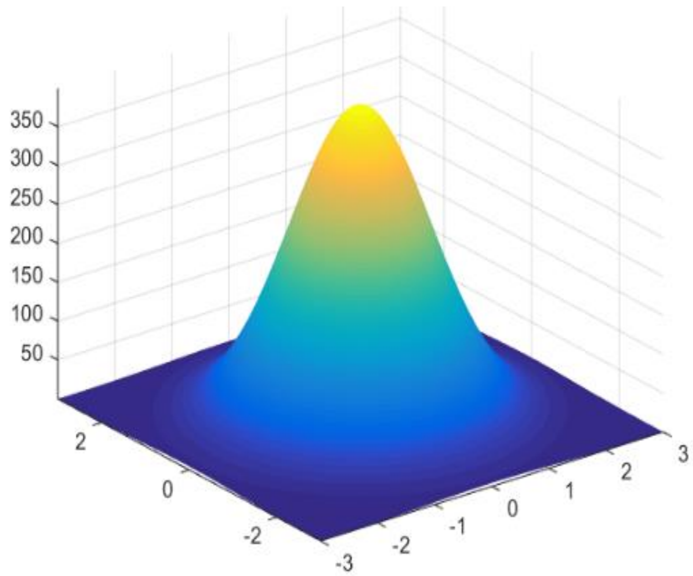
Multivariate distribution is the joint distribution that consists of multiple univariate random variables.

If we have a *p* x 1 random vector X that is distributed according to a multivariate normal distribution with population mean vector μ and population variance-covariance matrix Σ, then this random vector, X, will have the joint density function as shown in the expression below:

$$f(z;\mu,\Sigma) = (2\pi)^{-\left(\frac{N}{2}\right)} \det(\Sigma)^{-\frac{1}{2}} \exp\left(-.5(z-\mu)'\Sigma^{-1}(z-\mu)\right)$$

| Σ | denotes the determinant of the variance-covariance matrix Σ and Σ−1is just the inverse of the variance-covariance matrix Σ. Again, this distribution will take maximum values when the vector X is equal to the mean vector μ and decrease around that maximum.

If *p* is equal to 2, then we have a bivariate normal distribution, and this will yield a bell-shaped curve in three dimensions.



Bayes Gaussian Classifier

Bayes classification A better way is to assign the label based on the posterior probabilities (i.e., probabilities that a new data point belongs to the classes after we see it):

j =argmax(j) P (x 0∈ Cj | x)

According to Bayes' Rule, the posterior probabilities are given by

P (x ∈ Cj | x) = f (x | x ∈Cj) ·P (x ∈ Cj) f(x)

Therefore, the Bayes classification rule can be stated as

$$\boxed{\hat{j} = \text{argmax}_j \underbrace{f_j(\mathbf{x})}_{\text{likelihood}} \underbrace{\pi_j}_{\text{prior prob}}} \longleftarrow \text{generic Bayes classifier}$$

Confusion Matrix

C11: Test samples predicted as class 1 & actually belongs to class 1
C12: Test samples predicted as class 1 & actually belongs to class 2
C13: Test samples predicted as class 1 & actually belongs to class 3
C21: Test samples predicted as class 2 & actually belongs to class 1
C22: Test samples predicted as class 2 & actually belongs to class 2
C23: Test samples predicted as class 2 & actually belongs to class 3
C31: Test samples predicted as class 3 & actually belongs to class 1
C32: Test samples predicted as class 3 & actually belongs to class 2
C33: Test samples predicted as class 3 & actually belongs to class 3

Actual Class

	Class 1	**Class 2**	**Class 3**
Class 1	**C11**	**C12**	**C13**
Class2	**C21**	**C22**	**C23**
Class 3	**C31**	**C32**	**C33**

Accuracy = (C11+C22+C33) / Total number of samples used for testing.

Mean Precision: Number of samples correctly classified as positive class, out of all the examples classified as positive class.

Class1 precision = C11/C11+C21+C31
Class1 precision = C22/C12+C22+C32
Class1 precision = C33/C13+C23+C33

Recall: Number of samples correctly classified as positive class, out of all the examples belonging to positive class.

Class1 recall = C11/C11+C12+C13
Class1 recall = C22/C21+C22+C23
Class1 recall= C33/C31+C32+C33

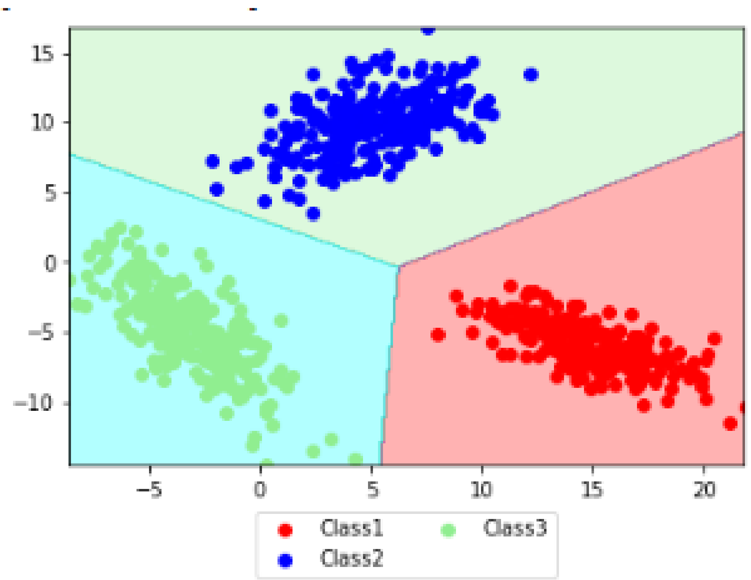
F-Measure:
Harmonic mean of precision and recall.

Class1 FMeasure = 2 Precision (Class1) * Recall (Class1) / Precision (Class1) +Recall (Class1)
Class2 FMeasure = 2 Precision (Class2) * Recall (Class2) / Precision (Class2) +Recall (Class2)
Class2 FMeasure = 2 Precision (Class3) * Recall (Class3) / Precision (Class3) +Recall (Class3)

Case 1: Covariance for all classes is I_2. Use the average of the sample variances for all dimensions, for all classes, from the training data as σ^2.

$$\Sigma_1=\Sigma_2=\Sigma_3=\Sigma= \sigma^2 I$$

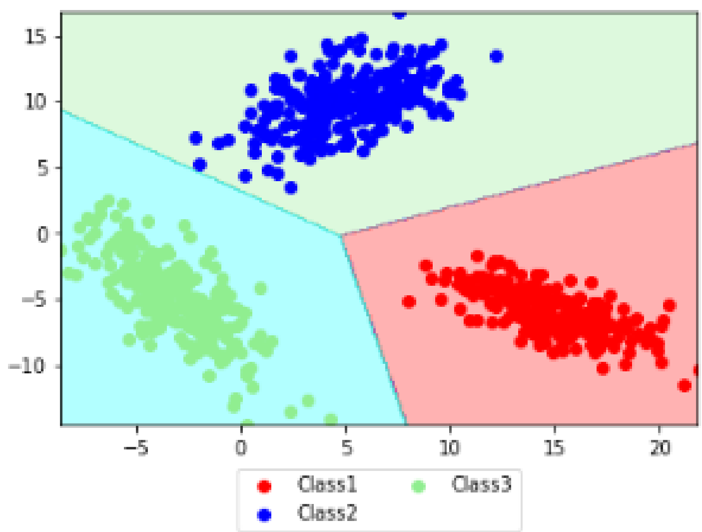
Classifier	Acc	Prec	Recall	Fscore
C1	100	1	1	1



Case 2: Full but equal covariance for all classes Σ. Use the average of the sample covariance matrix from all classes in the train data as Σ

$$\Sigma = (\Sigma 1 + \Sigma 2 + \Sigma 3)/3$$

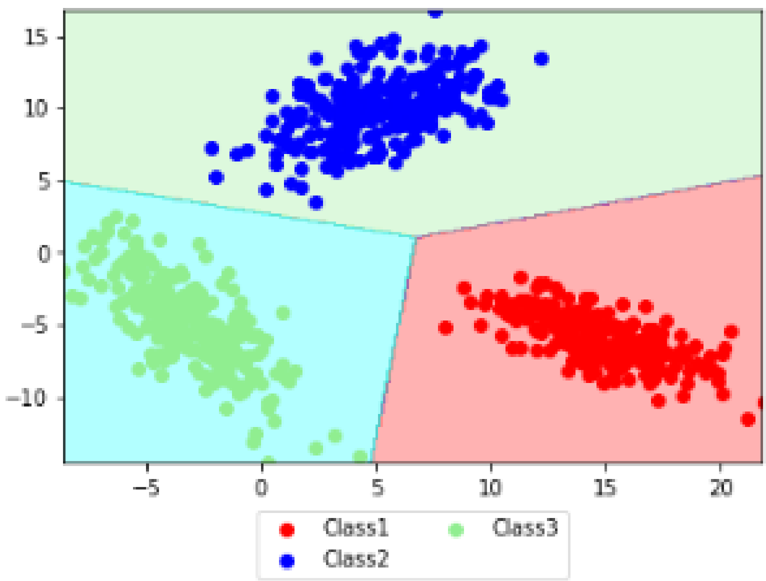
Classifier	Acc	Prec	Recall	Fscore
C2	100	1	1	1



Case 3: Diagonal covariance matrix, distinct for each class. Use variances from the sample covariance matrix for each class.

$$\Sigma 1 = \text{diagonal}(\sigma^2)$$
$$\Sigma 2 = \text{diagonal}(\sigma^2)$$
$$\Sigma 3 = \text{diagonal}(\sigma^2)$$

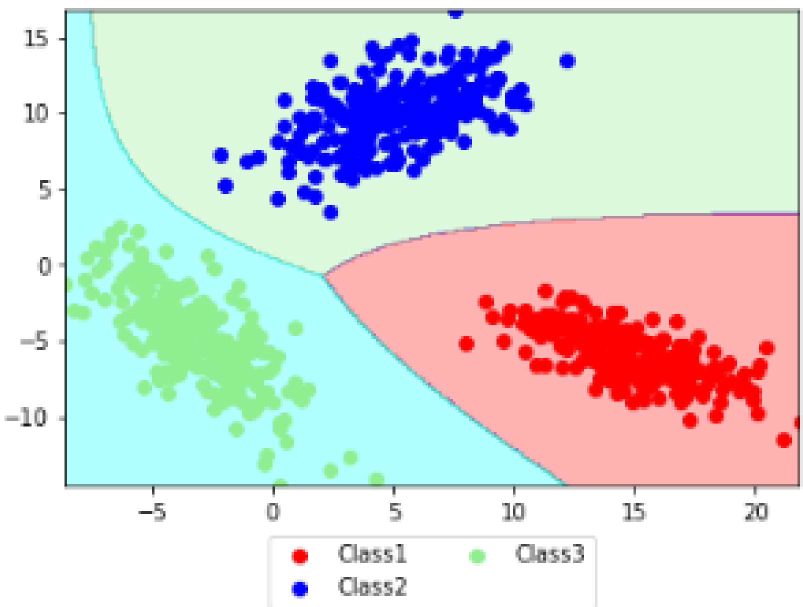
Classifier	Acc	Prec	Recall	Fscore
C3	100	1	1	1



Case 4: Full covariance matrix, distinct for each class. Use the sample covariance matrix for each class.

Σ1, Σ2, Σ3

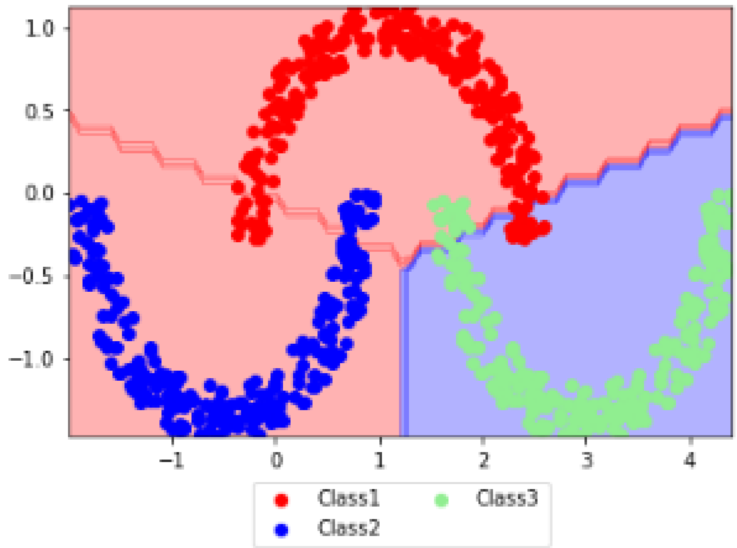
Classifier	Acc	Prec	Recall	Fscore
C4	100	1	1	1



Non-Linear Separable Case:

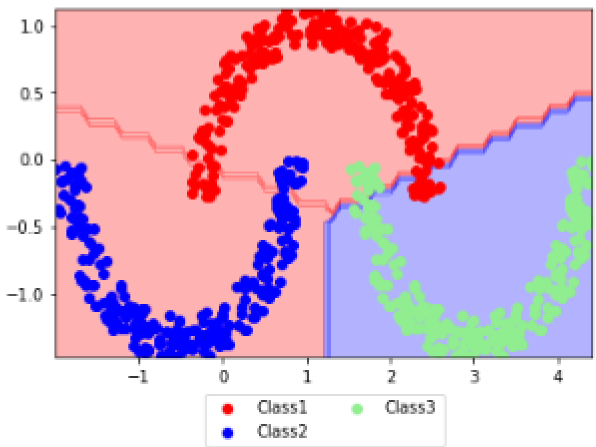
Case 1: Covariance for all classes is I_2. Use the average of the sample variances for all dimensions, for all classes, from the training data as σ^2.

Classifier	Acc	Prec	Recall	Fscore
C1	91.2	0.912	0.913181	0.912392



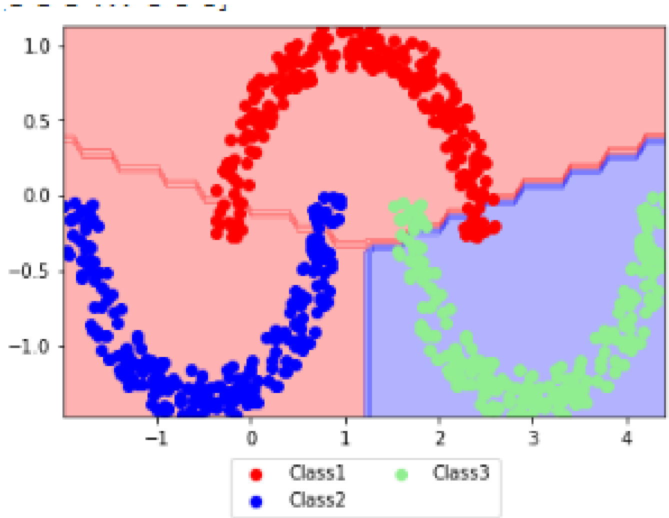
Case 2: Full but equal covariance for all classes Σ. Use the average of the sample covariance matrix from all classes in the train data as Σ

Classifier	Acc	Prec	Recall	Fscore
C2	91.2	0.912	0.913181	0.912392



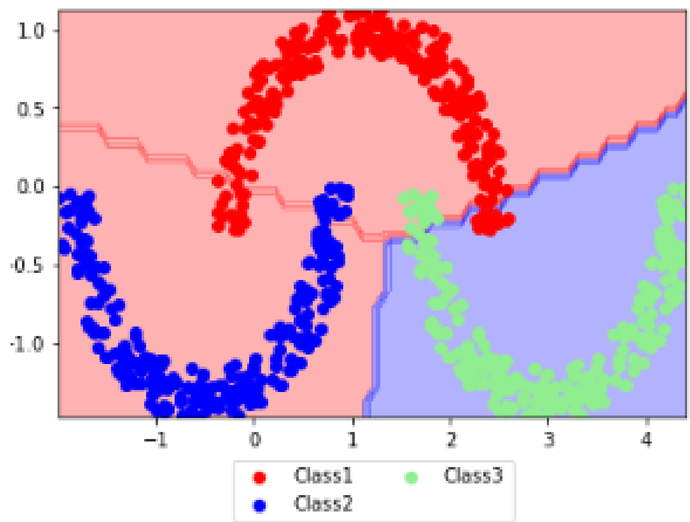
Case 3: Diagonal covariance matrix, distinct for each class. Use variances from the sample covariance matrix for each class

Classifier	Acc	Prec	Recall	Fscore
C3	91.6	0.916	0.916235	0.91607



Case 4: Full covariance matrix, distinct for each class. Use the sample covariance matrix for each class.

Classifier	Acc	Prec	Recall	Fscore
C4	92	0.92	0.920364	0.91997



Result for Linearly Separable Case

Classifier	Acc	Prec	Recall	Fscore
C1	100	1	1	1
C2	100	1	1	1
C3	100	1	1	1
C4	100	1	1	1

Result for Non-Linearly Separable Case

Classifier	Acc	Prec	Recall	Fscore
C1	91.2	0.912	0.913181	0.912392
C2	91.2	0.912	0.913181	0.912392
C3	91.6	0.916	0.916235	0.91607
C4	92	0.92	0.920364	0.91997